#6

Patent
Attorney's Docket No. 031941-096

jc971 U.S. PTO
09/993843
11/05/01

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of Klas NORDSTRÖM et al.	Group Art Unit: Unassigned
Application No.: Unassigned	Examiner: Unassigned
Filed: November 5, 2001	
For: Telecommunications Network Resource Handling Arrangement and Method	

"Express Mail" mailing label No. EL766105385US

Date of Deposit 11/05/01

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

J. Snead
(Type or printed name of person mailing paper or fee)

(Signature of person mailing paper or fee)

CLAIM FOR CONVENTION PRIORITY

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

The benefit of the filing date of the following prior foreign application in the following foreign country is hereby requested, and the right of priority provided in 35 U.S.C. § 119 is hereby claimed:

Sweden Patent Application No. 9901588-5

Filed: May 4, 1999

In support of this claim, enclosed is a certified copy of the prior foreign application. The prior foreign application is referred to in the oath or declaration. Acknowledgment of receipt of the certified copy is requested.

Respectfully submitted,

BURNS, DOANE, SWECKER & MATHIS, L.L.P.

Date: November 5, 2001

By: ______________________
Theodosios Thomas
Registration No. 45,159

P.O. Box 1404
Alexandria, Virginia 22313-1404
(919) 941-9240

PRV

PATENT- OCH REGISTRERINGSVERKET
Patentavdelningen

PATENT- OCH REGISTRERINGSVERKET

Intyg
Certificate

Härmed intygas att bifogade kopior överensstämmer med de handlingar som ursprungligen ingivits till Patent- och registreringsverket i nedannämnda ansökan.

This is to certify that the annexed is a true copy of the documents as originally filed with the Patent- and Registration Office in connection with the following patent application.

(71) Sökande *Telefonaktiebolaget L M Ericsson (publ), Stockholm SE*
Applicant (s)

(21) Patentansökningsnummer *9901588-5*
Patent application number

(86) Ingivningsdatum *1999-05-04*
Date of filing

Stockholm, 2001-10-11

För Patent- och registreringsverket
For the Patent- and Registration Office

Christina Vängborg

Christina Vängborg

Avgift
Fee *170:-*

E29 P70SE, 1999-04-28

Title:

Telecommunications network resource handler

TECHNICAL FIELD

The present invention relates to a resource handler for use in an operational support structure for managing a telecommunications network, comprising a service and resource database containing information regarding network resources. The invention also relates to a method of structuring information in a resource handler database for use in such an operational support structure. Also, the invention relates to a use of a resource handler for a service type handler in an operational support structure for a telecommunications network, for creating and maintaining service type recipes and their relations.

STATE OF THE ART

Resource management in operation support systems for tele-communication is difficult due to the size and complexity of these types of networks. A telecommunication network normally changes constantly as new resources/services are added, and as old superfluous resources/services are removed or replaced. Often, a lack of information regarding available/redundant resources lead to stove-pipe solutions for particular services or networks with high levels of duplication. The lack of up-to-date information and the resulting duplication lead to higher management costs for the network operator. Sometimes resource investments are unnecessary because redundant resources may actually be available. There may be resources that supposedly are engaged by old services that are

no longer active or have been replaced by new services that use other resources.

An operational support structure for a telecommunications network is for example described in US 5,640,505. This support structure is divided into a set of domains each of which provides a particular management function for the network. This document teaches a general support structure but it provides very little details for implementing such a structure. Thus, there is no definite solution to the problem of how to handle network resources so that duplication is avoided.

Ideally, the service and resource database should be able to represent all existing and all future types of telecommunication equipment and network topologies in a simple data model in order to be able to manage the resources of an arbitrary network.

SUMMARY OF THE INVENTION

According to a first aspect of this invention, there is provided a resource handler for use in an operational support structure for managing a telecommunications network, comprising a service and resource database containing information regarding network resources. The database is structured so that each resource in the network has a time of existence as well as a place in a hierarchy of parent/child(s) relations. The resource is defined by the following data:

a point identifier that has characteristics associated to it, in the form of an abstract description of its capabilities;

an abstraction of the common network element in the sense of a group of points that are considered to belong together; and

a connection which is defined by two connected points.

Preferably, the point identifier also has characteristics associated to it, in the form of a list of label/value pairs.

According to one advantageous embodiment of the invention, the element acts as a container for points, with the implicit characteristic that points on elements are possible to cross-connect.

The database may be structured so as to model a topological view, i.e. how the resources are connected together. It may also be structured so as to model a time view, i.e. when the resources exist.

The database may be structured so as to model a hierarchic view, i.e. how the resources are related in parent/child relationships. It may also be structured so as to model a characteristic view, i.e. by means of a list of characteristics of each resource. Also, the database is structured so as to model a usage view, i.e. which resources are combined to form a complete service instance and the time when that service instance exists.

Preferably, the topological view, the time view, the hierarchic view, the characteristic view and the usage view are integrated in a data model for enabling control of each resource and the use of it in service instances.

According to a second aspect of this invention, there is provided a method of structuring information in a resource handler database for use in an operational support structure for managing a telecommunications network, comprising the steps of allocating

each resource in the network a time of existence as well as a place in a hierarchy of parent/child(s) relations, and

defining each resource by the following data:

a point identifier that has characteristics associated to it, in the form of an abstract description of its capabilities;

an abstraction of the common network element in the sense of a group of points that are considered to belong together; and

a connection which is defined by two connected points.

According to a third aspect of this invention, there is provided a use of a resource handler for a service type handler in an operational support structure for a telecommunications network, for creating and maintaining service type recipes and their relations.

Preferably, the service type recipes provides a framework for service types, operations on service types, parameters on service types, hierarchical relations between service types, hierarchical parameter relationship, and translation of service types and associated parameters values into resource requirements and service type requirements.

The service type handler may be used for selecting between different types of required services, different types of required resources and different service instances.

Advantageously, the selected resources requirements are transferred to a resource handler that does the actual resource allocation.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will in the following be further described in a non-limiting way with reference to the accompanying drawings in which:

FIG. 1 schematically illustrates the relationship between the concepts of product instance/type and resource instance/type,

FIG. 2 schematically illustrates the relationship between elements, points and connections,

FIG. 3 shows the system environment of the service and resource database of the service configuration,

FIG. 4 is a block diagram which illustrates the relational structure of a service and resource database, and

FIG. 5 illustrates the table relations and product type definitions.

DETAILED DESCRIPTION OF THE INVENTION

As there are many aspects of a telecommunications system, a structure of the information has to be made into parts for interaction between product types and product instances. For example, it is possible to have one part of the system handling product types and their translation to product instances and/or references to resources, a second part for binding information about product types and product type pricing, a third part for the product instances and a fourth part for keeping track of all procedural code for data manipulation.

This basic modeling idea is illustrated in FIG. 1, and it comprises four basic areas (domains), the product type, the product instance, the resource type and the resource instance. The type areas are to be seen as where the drawings and design specifications are placed and the instance areas are where the raw

material and the ready products are located. Thus, it is very important to make a clear distinction between the type of a product and the instance of a product as well as between a resource type and a resource instance.

Each product type is defined by name, parameters, revision level and behavior. Thus the product type can be viewed as a parameterized template for a product instance. Each product type has one or more revisions. As a product type may consist of other product types, some product types are placed in a hierarchy of parent/child relations. It is of outmost importance that a product type has no knowledge of the product types it is part of. It may know things about the product types it consists of. One parent product type may have the possibility to fork to one or several child product types.

A product instance is the realization of a product with all parameter values set to their final values. Since the product types have a hierarchy, the product instances also must have a hierarchy. The product process is the procedure referenced to by the product type. It contains the precise and complete instructions how to manipulate a product instance.

The concept of resources is used in order to have a concept for all "real" things in the system. It is very important to distinguish between two things, resource creation, i.e. installation of a new telephone line, and resource allocation, i.e. changing the state of an existing telephone line from unused to used.

The resource type is the identity of an abstract description of the resource needed to implement a product type. Product types having children may have references to resource types. Product types having no children must have references to resource types.

The resource instance is the actual instantiation of a resource type. Product instances having children may have references to resource instances. Product instances having no children must have references to resource instances.

The relationship between elements, points and connections is illustrated in Fig. 2 which is an abstract graphic view of a telecommunication network. Each of the squares in Fig. 2 represents an element 10, the dots are points 11 and the fat lines are connections 12. The thin lines in the lower left element are also connections, only scaled down in width to fit the element. This is represented in the Service Resource Data Base (SRDB) according to this invention, in the form of tables in a relational database.

Having an SRDB is to have a formal and abstract model for describing a telecommunication network, its parts, their capabilities, how the parts are connected, by whom and when a part is used and so on. What exists in SRDB are lists of elements, the points on these elements, the capabilities/attributes of each point and how the points are connected. These lists (in the form of database tables) are filled with information of what equipment exists (and when it exists), what it can do and if it is used by someone.

The point 11 is known under many names, such as port, termination point, external access point, binding post etc. in the real world. In SRDB they are abstracted to one concept, the point which in essence is an identifier. To distinguish between different types of points, every point must have a type that is a formal abstract description of its capabilities. A point type may have one or several capabilities. A point must always belong to an element, it can not exist on its own.

The element 10 is in essence a group of points that are considered to belong together. To be an element, the points that are attached to it must have full connectivity to each other, i.e. any point must be possible to cross connect to any other point within the same element. This connectivity may be restrained a bit as the capability/attribute types of the points may have the requirement that the same capability/attribute must exist on both points in order to be able to cross connect.

The connection is the very information that says that a point is connected to another point. In reality this is a pair of points. There are basically two kinds of connections, the infrastructure connection, i.e. a wire, and a cross connect, which is controllable.

The capability is a collection of attributes. Each attribute is in essence a pair consisting of the name of a parameter and the value of the parameter. This is the formal way to describe the characteristics of a point type.

The service instance is the entity that in itself has all the common data regarding the resources used by it and it has references to these resources.

The concept of hierarchy is used in many places in SRDB in the form that elements 10, points 11 and connections 12 may have children, i.e. they may have an internal structure. This is described in the form of a parent/child relationship in a table.

Fig. 3 shows the system environment where the Service Configuration (SC) mainly consist of the activation engine 13, the Service Request Processor (SRP) 14 for upstream system communication, the Network Element Processor (NEP) 15 for downstream system communication, Service Resources Database (SRDB) 16 for information about the network resources, and Service Type Database (STDB) 17, see FIG. 5, for the service type hierarchy. Internally, the activation engine 13 consist of Service Activation Request Manager (SARM) 18 a part of an Automatic Service Activation Program (ASAP) and the Service and Resource Instance Manager (SRIM) 19.

The STDB 17, see FIG. 5, contains all information on how to translate a specific service type to resources, to enable creation of new service instances. The SRDB 16 is the information source and depository of SC.

A simplified view of the flow of information is that a Common Service Description Layer (CSDL) command is sent by the SRP 14 to SARM 18 where it is translated to an Atomic Service Description Layer ASDL command and routed to SRIM 19. SRIM then asks STDB 17 and SRDB for additional information which is used to make a new

CSDL command is translated to an ASDL command and routed to the NEP 15. The response from the NEP follows in principle the same way backwards, where return status and data is used to update SRDB 16.

The creation of a DuoCom instance is now used as an example of how to use the information in the data structures. The product DuoCom is assumed to consist of the sub products ISDN Access, E-mail, Personal Home Page and 020-connection. These five products, together with all relevant parameters, parameter values, resource definitions, location of resources, product information, product type cost, revision levels etc. and the associated provisioning control are considered to be entered in the product type data structure.

Step zero is that an initiator, probably a customer care system (at some convenient time) has got access to an Operator Product Portfolio (OPP) and knows at least the names of the available products. When an end user contacts the customer care system, the customer is given a list of product types to select from. In this case the DuoCom product is the interesting one, so a request is made for more information about the product type.

In step 1 the OPP, by using the product name and the fact that product information is wanted as parameters, looks in the product type information table where all aspects of information and where to find it is located.

In step 2, assuming that the customer wants the product, a feasibility check is made to check if it is possible to implement the product with the existing conditions, i.e. if all

prerequisites are fulfilled. Now a "call" for this code is made. As the execution starts, this code uses the product type is input to look in the Product type hierarchy. By looking here it is possible to find out which (sub)products the DuoCom product is composed of. In this case the four product types ISDN Access, E-mail, Personal Homepage and 020-connection will be found. As they in turn may consist of further more simple products, the Product type hierarchy is once again examined to see if the new products have sub products and so on until no more product types are found.

As some of the (sub)products may be optional, i.e. the end customer has to be asked the (sub)product is desired and informed about the possible choices. This results in an interactive loop in which the customer picks the desired (sub)products. As these (sub)products are selected, the Product type operation parameters and the Product type parameter tables are examined to find out which parameters are needed. As this selection of (sub)products is dynamic to its nature, the Product type relations table is examined to check that combinations of incompatible product types are not accidentally created. When this loop has ended, the Product type prerequisites and the Product type resource prerequisites tables are examined with regard to Product instance data and resource data, if the sufficient amount of resources or existing product instances are available, so that it is possible to instantiate this new product instance.

In step 3 the Product type cost and Product type hierarchy tables are examined to get the cost for all individual (sub)products. It is however left to the customer care system to interpret and customize this information for the customer. This price system may

be in an internal currency and the customer care system may transform this currency into a local currency.

In step 4, assuming that the feasibility showed that implementation was possible, the created instance order may be issued, i.e. to take all data from the feasibility and create one or several product instantiation orders. This will insert one or more product orders in Product order tables and result in a tree of product instances in the Product instances tables.

In step 5, when the time of delivery arrives, it is finally up to the provisioning processor to activate the product, i.e. traversing the product instances in the product instance hierarchy and to read out all data and give orders to the underlying resource managers to activate the reserved resources.

All data is stored in a relational database, see FIG. 4 which shows only the table names and the fields in each table that has relations to/from them. The tables may be created by reading from standard text files with SQL commands.

CLAIMS

1. A resource handler for use in an operational support structure for managing a telecommunications network, comprising a service and resource database containing information regarding network resources,

c h a r a c t e r i z e d in

that the database is structured so that each resource in the network has a time of existence as well as a place in a hierarchy of parent/child(s) relations, and

that the resource is defined by the following data:

a point (11) identifier that has characteristics associated to it, in the form of an abstract description of its capabilities;

an abstraction of the common network element (10) in the sense of a group of points (11) that are considered to belong together; and

a connection (12) which is defined by two connected points (11).

2. A resource handler according to claim 1, c h a r a c t e r i z e d in that the point (11) identifier also has characteristics associated to it, in the form of a list of label/value pairs.

3. A resource handler according to claim 1 or 2, c h a r a c t e r i z e d in that the element (10) acts as a container for points (11), with the implicit characteristic that points (11) on elements (10) are possible to cross-connect.

4. A resource handler according to any one of claims 1 to 3, characterized in that the database is structured so as to model a topological view, i.e. how the resources are connected together.

5. A resource handler according to any one of claims 1 to 4, characterized in that the database is structured so as to model a time view, i.e. when the resources exist.

6. A resource handler according to any one of claims 1 to 5, characterized in that the database is structured so as to model a hierarchic view, i.e. how the resources are related in parent/child relationships.

7. A resource handler according to any one of claims 1 to 6, characterized in that the database is structured so as to model a characteristic view, i.e. by means of a list of characteristics of each resource.

8. A resource handler according to any one of claims 1 to 7, characterized in that the database is structured so as to model a usage view, i.e. which resources are combined to form a complete service instance and the time when that service instance exists.

9. A resource handler according to claims 4-8, characterized in that the topological view, the time view, the characteristic view and the usage view, the hierarchic view, the characteristic view and the usage view are integrated in a data model for enabling control of each resource and the use of it in service instances.

10. A method of structuring information in a resource handler database for use in an operational support structure for managing a telecommunications network, comprising a service and resource database containing information regarding network resources, characterized in the steps of

allocating each resource in the network a time of existence as well as a place in a hierarchy of parent/child(s) relations, and

defining each resource by the following data:

a point (11) identifier that has characteristics associated to it, in the form of an abstract description of its capabilities;

an abstraction of the common network element (10) in the sense of a group of points (11) that are considered to belong together; and

a connection (12) which is defined by two connected points (11).

11. A method according to claim 10, characterized in the step of associating the point (11) identifier with characteristics, in the form of a list of label/value pairs.

12. A method according to claim 10 or 11, characterized in the step of allowing the element (10) to act as a container for points (11), with the implicit characteristic that points (11) on elements (10) are possible to cross-connect.

13. A method according to any one of claims 10 to 12, characterized in the step of structuring the database

so as to model a topological view, i.e. how the resources are connected together.

14. A method according to any one of claims 10 to 13, characterized in the step of structuring the database so as to model a time view, i.e. when the resources exist.

15. A method according to any one of claims 10 to 14, characterized in the step of structuring the database so as to model a hierarchic view, i.e. how the resources are related in parent/child relationships.

16. A method according to any one of claims 10 to 15, characterized in the step of structuring the database so as to model a characteristic view, i.e. by means of a list of characteristics of each resource.

17. A method according to any one of claims 10 to 16, characterized in the step of structuring the database so as to model a usage view, i.e. which resources are combined to form a complete service instance and the time when that service instance exists.

18. A method according to claims 13-17, characterized in the step of integrating the topological view, the time view, the hierarchic view, the characteristic view and the usage view in a data model for enabling control of each resource and the use of it in service instances.

19. A use of a resource handler according to anyone of claims 1 to 9 for a service type handler in an operational support structure for a telecommunications network, for creating and maintaining service type recipes and their relations.

20. A use according to claim 19, wherein the service type recipes provides a framework for service types, operations on service types, parameters on service types, hierarchical relations between service types, hierarchical parameter relationship, and translation of service types and associated parameters values into resource requirements and service type requirements.

21. A use according to anyone of claims 19-20, for selecting between different types of required services, different types of required resources and different service instances.

22. A use according to claim 21, wherein the selected resources requirements are transferred to a resource handler that does the actual resource allocation.

ABSTRACT

A resource handler for use in an operational support structure for managing a telecommunications network comprises a service and resource database containing information regarding network resources. The database is structured so that each resource in the network has a time of existence as well as a place in a hierarchy of parent/child(s) relations. The resource is defined by the following data: a point (11) identifier that has characteristics associated to it, in the form of an abstract description of its capabilities; an abstraction of the common network element (10) in the sense of a group of points (11) that are considered to belong together; and a connection (12) which is defined by two connected points.

(FIG. 2)

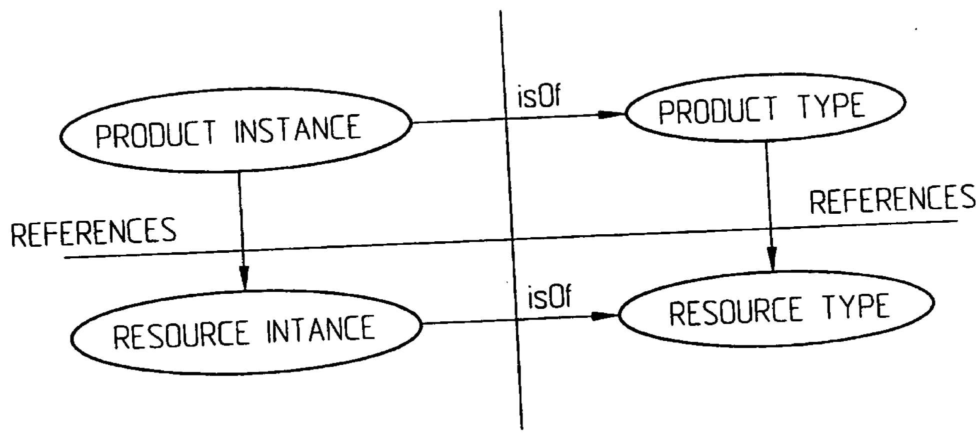

PRODUCT INSTANCE
isOf
PRODUCT TYPE
REFERENCES
REFERENCES
RESOURCE INTANCE
isOf
RESOURCE TYPE

Fig. 1

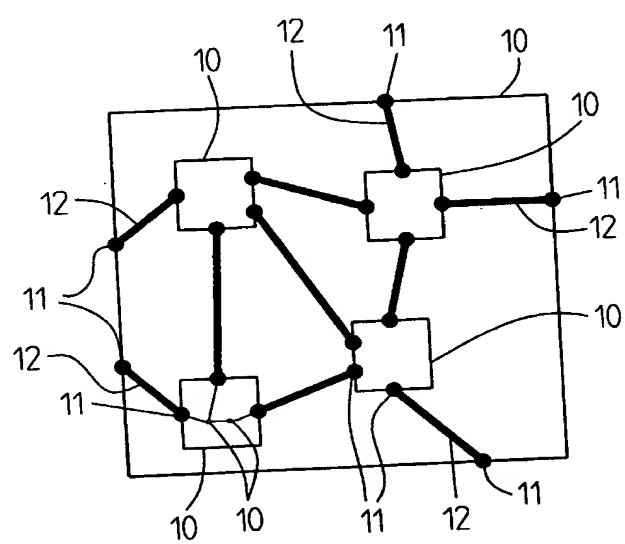
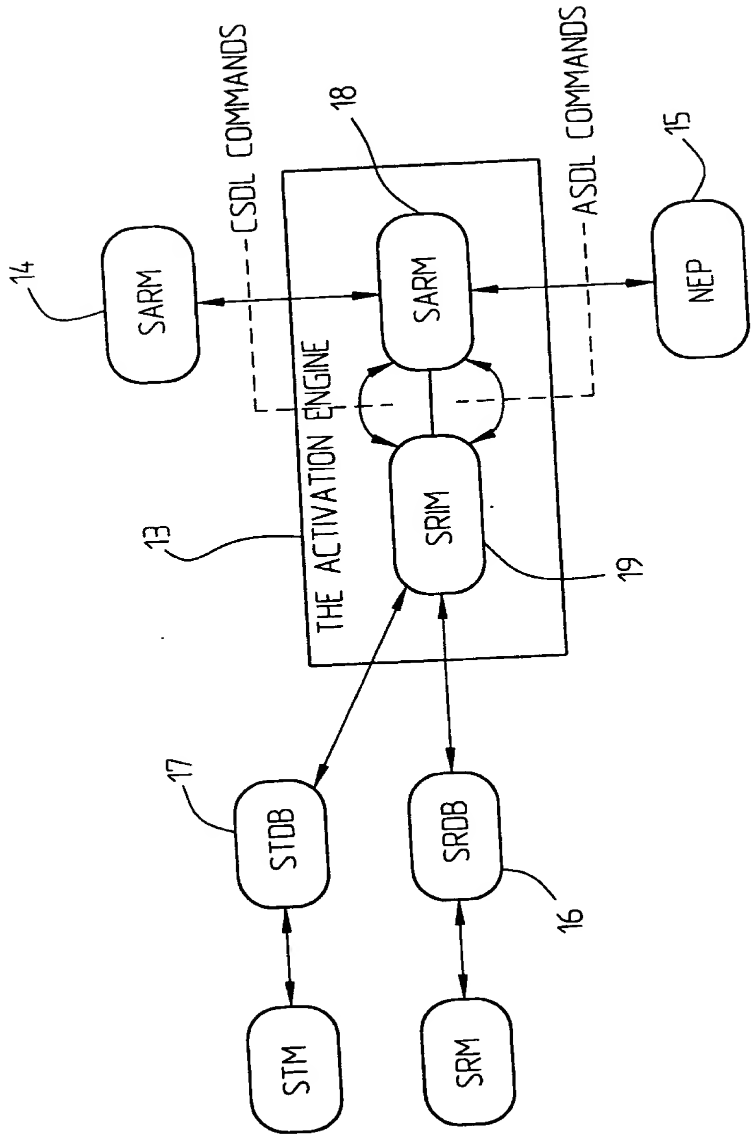

12
11
10
10
10
12
11
12
11
11
12
10
11
10
10
11
12
11

Fig. 2



THE ACTIVATION ENGINE
SARM
SARM
NEP
SRIM
STDB
SRDB
STM
SRM
CSDL COMMANDS
ASDL COMMANDS
13
14
15
16
17
18
19

Fig. 3

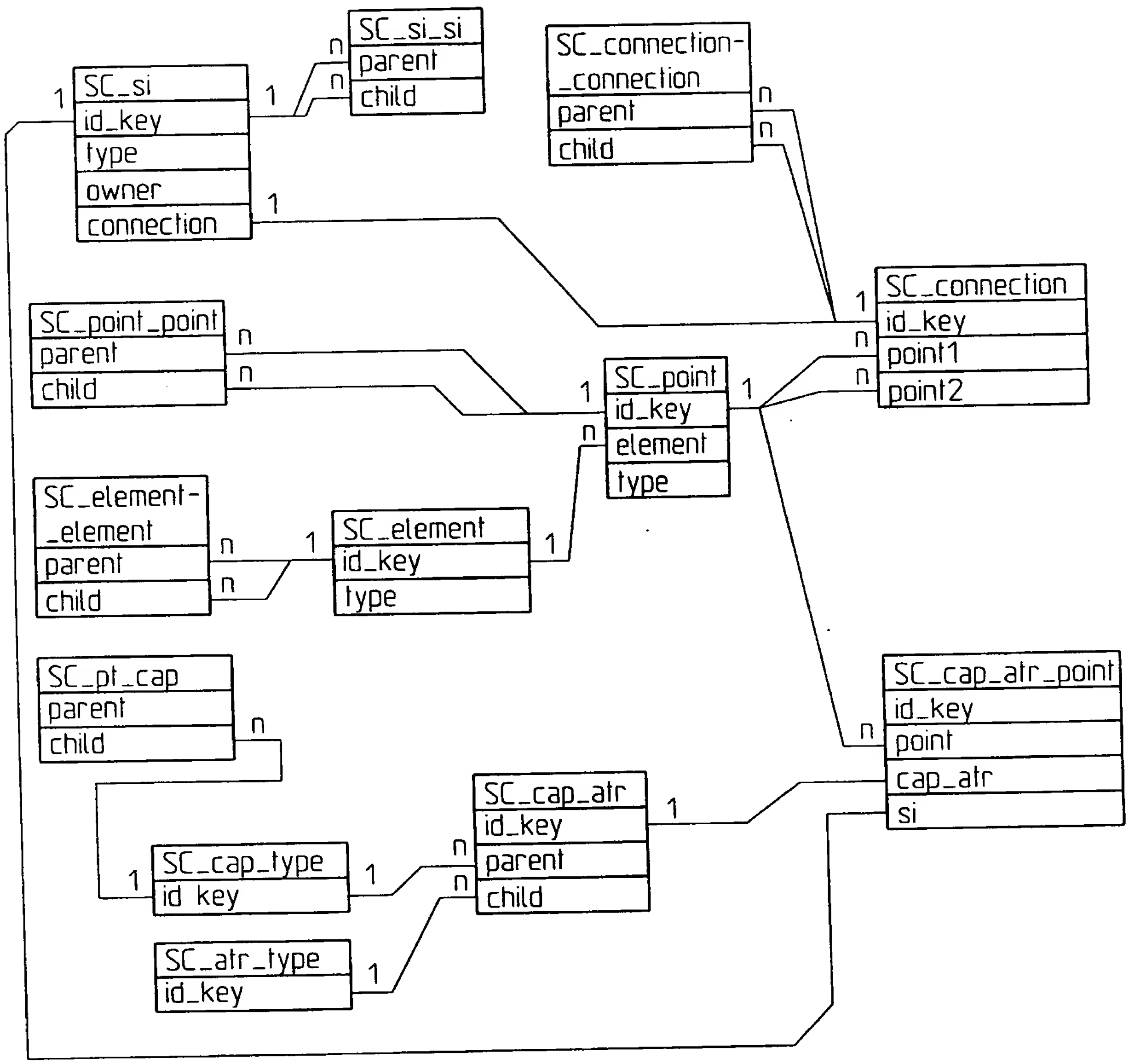

SC_si
id_key
type
owner
connection
SC_si_si
parent
child
SC_connection-
_connection
parent
child
SC_connection
id_key
point1
point2
SC_point_point
parent
child
SC_point
id_key
element
type
SC_element-
_element
parent
child
SC_element
id_key
type
SC_pt_cap
parent
child
SC_cap_atr_point
id_key
point
cap_atr
si
SC_cap_atr
id_key
parent
child
SC_cap_type
id key
SC_atr_type
id_key
1
n

Fig. 4

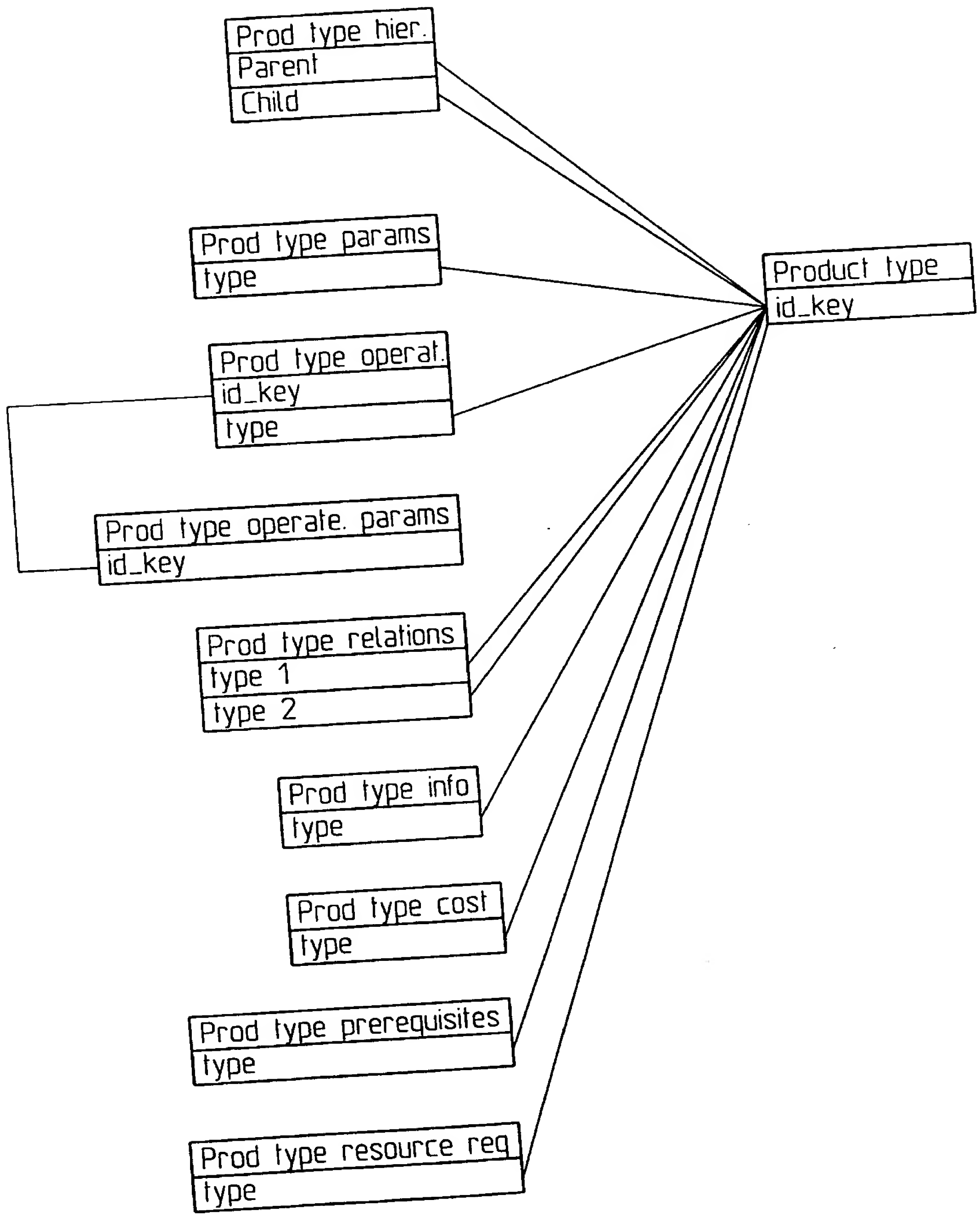

Prod type hier.
Parent
Child
Prod type params
type
Prod type operat.
id_key
type
Prod type operate. params
id_key
Prod type relations
type 1
type 2
Prod type info
type
Prod type cost
type
Prod type prerequisites
type
Prod type resource req
type
Product type
id_key

Fig.5